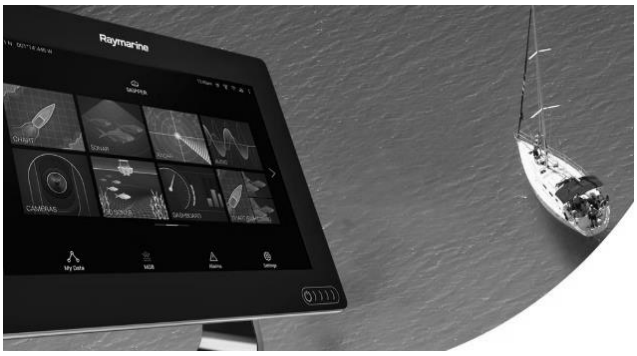
Elektronik in Yachten

Aufgaben



https://www.blauwasser.de/marke/raymarine (abgerufen am 15.8.2021).

Moderne Yachten verfügen über zahlreiche vernetzte Systeme an Bord, die die Crewmitglieder unterstützen und für die Sicherheit an Bord sorgen. Zu diesen Systemen zählen u. a. die Geber (Messsensoren) für die Windgeschwindigkeit, die Windrichtung, die Geschwindigkeit durch das Wasser, die Wassertiefe und die GPS-Position. Als Aktor wird beispielsweise das mit einem Motor angetriebene Steuerrad eingesetzt, mit dem der Kurs der Yacht verändert werden kann. Auch die Signalbeleuchtungen, wie z. B. das Ankerlicht und die Positionsbeleuchtung, sind Aktoren. Als Kommunikationssysteme mit dem Skipper dienen verschiedene Displays. Eine Möglichkeit der Vernetzung ist es, eine Datenzentrale an Bord zu haben und mittels Client-Server-Verbindungen mit den Sensoren, Aktoren und den Displays die Daten protokollgestützt auszutauschen.

1 Das Einrichten der Datenzentrale wird von einer Servicetechnikerin oder einem Servicetechniker (im folgenden Servicetechniker genannt) vorgenommen. Hierzu muss er sich an der Datenzentrale anmelden, wobei er den Usernamen und das Passwort eingeben muss. Beim Einrichten werden die auf den Sensoren und Aktoren aufgedruckten Geräteadressen sowie die Gerätetypen bei der Datenzentrale eingegeben. Einige Geräte müssen kalibriert werden, andere nicht. Außerdem kann der Servicetechniker ein Protokoll der an der Datenzentrale angeschlossenen Geräte aufrufen und es gegebenenfalls ausdrucken. Das Crewmitglied kann Bildschirmgeräte, die an der Datenzentrale angeschlossen sind, konfigurieren, d. h. den Bildschirm in verschiedene Fenster unterteilen, zuordnen, welche Daten in den jeweiligen Fenstern angezeigt werden sollen, sowie Warnschwellen setzen, ab denen ein akustisches Signal ausgesendet werden soll.

1.1 Entwickeln Sie unter Berücksichtigung der obigen Systembeschreibung ein Use-Case-Diagramm und bestimmen Sie die Akteure, Beziehungen zwischen Akteuren und Anwendungsfällen und eventuell vorhandene include- bzw. extend-Beziehungen.

(12 BE)

1.2 Eine Yacht besitzt eine Datenzentrale. Für das Einrichten einer Datenzentrale muss man sich an der Datenzentrale einloggen und den Usernamen und das Passwort eingeben. Es können Geräte hinzugefügt und gelöscht werden. Es gibt verschiedene Geräte, die an eine Datenzentrale angeschlossen werden können. Alle Geräte haben eine Geräteadresse und einen Gerätetyp. Jedes Gerät muss auf seine Datenzentrale zugreifen können.
Es gibt Windsensoren, die die Windrichtung und die Windgeschwindigkeit messen können. Die Positionsleuchten in den Farben rot, grün oder weiß können eingeschaltet und ausgeschaltet sein. Man kann den Modus zwischen eingeschaltet und ausgeschaltet wechseln. Zudem gibt es einen Steuerradmotor, der die Modi manuell und Autopilot einnehmen kann. Der Modus kann

zwischen „Autopilot eingeschaltet“ und „Autopilot ausgeschaltet“ gewechselt werden. Wenn der Autopilot eingeschaltet ist, ermittelt die Navigationseinrichtung aus der Differenz zwischen Soll-Kurs und Ist-Kurs das Steuersignal für den Steuerradmotor.
Modellieren Sie aus dem oben dargestellten Sachverhalt ein UML-Klassendiagramm.
Stellen Sie dabei alle benötigten Attribute und deren Datentypen, Assoziationen, Multiplizitäten, eventuelle Vererbungsstrukturen, Sichtbarkeiten und abstrakte Klassen dar.

Hinweis: Es sind auch die genannten Operationen zu modellieren. Konstruktoren, Setter- und Getteroperationen müssen nicht dargestellt werden.

(13 BE)

1.3 Eine Navigationseinrichtung besteht aus der Klasse `Navigation` und einem angeschlossenen GPS-Sensor (Material 1). Der GPS-Sensor ermittelt kontinuierlich Positionsdaten und speichert diese in seinem Attribut `position`. Die Operation `positionAktualisieren()` liefert die aktuelle Position zurück. In der Klasse `Navigation` wird die Liste mit Positionsdaten `positionsListe` gehalten, die zyklisch in der Operation `zyklus()` um einen neuen Eintrag mittels der Operation `positionHinzufuegen()` erweitert wird. Die aktuelle Position in der Liste ist mittels `lfdNr` erreichbar, die davor mit `lfdNr-1`. Hier ist auf Zugriffsfehler zu achten. Ferner ist in der Methode `zyklus()` der `istKurs` zu setzen. Die Instanziierung des Objekts GPS wird in der Klasse `Navigation` vorgenommen.
Da Sie sorgfältig testen wollen, wird eine `TestUI` benötigt, in der Sie in der Operation `main()` die Navigationsklasse instanziieren und dreimal die Operation `zyklus()` aufrufen. Zu Testzwecken geben Sie am Ende der Operation `zyklus()` eine Testnachricht aus: „aktueller Kurs = … , Element in der positionsListe = …“.
Implementieren Sie das Klassendiagramm (Material 1) in der in Ihrem Unterricht verwendeten objektorientierten Programmiersprache.

Hinweise: Die Operation `istKursErmitteln()`, Setter- und Getter-Operationen sind nicht zu implementieren.

(15 BE)

2 Die Sensoren und Aktoren an Bord sind mit Mikrocontrollern ausgestattet. Auf der Mastspitze von Yachten befindet sich immer ein Ankerlicht, das eingeschaltet wird, wenn man nachts unter Anker liegt. Ebenfalls auf der Mastspitze angebracht ist ein Windsensor, der die Windgeschwindigkeit und die Windrichtung ermittelt. Diese Sensor-/Aktorkombination soll mit einem Mikrocontroller ausgestattet werden. In dem Mikrocontroller ist bereits eine Kommunikations-Software installiert, die die Kommunikation mit der Datenzentrale der Yacht durchführt. Die Kommunikations-Software benutzt die Register R1 und R2, um mit der von Ihnen zu entwickelnden Software zu kommunizieren.

2.1 In Material 2 ist das Windsensorsystem dargestellt. Oben befindet sich die Windfahne, die die Windrichtung gemessen zur Schiffsachse ermittelt und unten ist das Windrad, welches die Windgeschwindigkeit misst. Je Umdrehung des Windrads werden 4 Impulse ausgelöst, die gezählt werden müssen. Alle 30 Sekunden sollen für eine Messzeit von 4 Sekunden die Impulse gezählt und anschließend die Umdrehungen pro Sekunde (mathematisch gerundet) in dem Register R1 gespeichert werden.
Zum Zählen wird ein externer Zähler verwendet, der mit dem Mikrocontroller verbunden ist (Material 3). Die Steuersignale für den Zähler müssen Impulse mit einer Länge von 10 ms sein. Start dient dem Starten des Zählers, Stopp hält ihn an und mit Rueck wird der Zähler auf Null

zurückgesetzt. Der Zählerstand kann am Port D eingelesen werden. Die Kommunikations-Software setzt im Register R2 das Bit 0, wenn das Ankerlicht (Port B, Bit 0) eingeschaltet werden soll und löscht das Bit (Port B, Bit 0) für das Ausschalten des Ankerlichts.
Entwickeln und zeichnen Sie das Struktogramm für den hier beschriebenen Teil der Software.

Hinweis: Wenn Sie Ports und Bits anpassen wollen, sind diese Änderungen kenntlich zu machen.

(10 BE)

2.2 Implementieren Sie das in Aufgabe 2.1 beschriebene Vorgehen für den in Ihrem Unterricht verwendeten Mikrocontroller in Assembler.

Hinweis: Die Warteschleifen sind nicht zu implementieren. Für die benötigten Warteschleifen sind Unterprogrammaufrufe mit sinnvollen Bezeichnungen, z. B. `time2sek` für eine Warteschleife mit einer Dauer von 2 Sekunden, zu verwenden.

(12 BE)

2.3 Im Rahmen von Wartungsarbeiten muss der externe Zähler ausgetauscht werden. Der verbaute Zähler ist nicht mehr lieferbar und muss durch einen Zähler mit anderen Steuersignalen ersetzt werden. Es muss lediglich die Hardwareschaltung ergänzt werden, um den neuen Zähler zu steuern. Der neue Zähler besitzt nur den Rücksetzeingang rueck und einen Eingang count. Liegt am Eingang count eine 1 an, so zählt der Zähler, wenn eine 0 anliegt, hält der Zähler an.
Entwickeln und zeichnen Sie die Hardwareschaltung für die Integration des neuen Zählers in die vorhandene Schaltung.

Hinweis: Material 3 zeigt die ursprüngliche Schaltung mit dem alten Zähler.

(6 BE)

2.4 Alternativ zur obigen Hardwareschaltung sollen per Software die Impulse von winSpeed ermittelt werden, um daraus Rückschlüsse auf die Windgeschwindigkeit ziehen zu können. Dazu muss zunächst der Signalverlauf analysiert werden. Der Signalverlauf windSpeed des Windrads ist in Material 4 in einem Signal-Zeit-Diagramm dargestellt. Erläutern Sie einen möglichen Lösungsansatz, um die Impulse korrekt per Software mit einem Mikrocontroller zählen zu können, wenn sie auf den Einsatz interner Timer/Zähler verzichten.

(5 BE)

2.5 Material 5 zeigt ein anderes mögliches Signal-Zeit-Diagramm, in welchem zwei aufeinander folgende Impulse des Signals windSpeed dargestellt sind. Die Impulse können eine maximale Dauer von T_max haben. Die minimale Dauer zwischen dem Beginn des ersten Impulses und dem Beginn des zweiten Impulses beträgt T_min. Nennen Sie eine mögliche Ursache für das Auftreten dieses typischen Signalverlaufs und erläutern Sie Ihre Softwarestrategie zur Zählung der Impulse, ohne interne Zähler zu verwenden.

(5 BE)

2.6 Das Windrad in Material 2 gibt über einen Winkelkodierer die Winkelstellung der Windfahne aus. Drehgeber können mittels einer Codescheibe, die von IR-Lichtschranken abgetastet werden, die aktuelle Position in Grad des Drehgebers angeben. Hierbei lösen die dunklen Stellen eine 1 und die hellen Stellen eine 0 aus. In Material 6 sind zwei einfache Codescheiben dargestellt, die jeweils durch drei IR-Lichtschranken abgetastet werden. Die eine Code-Scheibe ist binärcodiert und die andere ist im Graycode codiert. Analysieren sie den Unterschied zwischen den beiden Codes und diskutieren Sie die Vorteile bei Verwendung des Graycodes.

(4 BE)

3 Das Modul aus Aufgabe 2, gebildet aus Windsensor und Ankerleuchte, ist über eine Client-Server-Verbindung mit der Datenzentrale verbunden. In Material 7 ist das RFC-Protokoll der Datenkommunikation wiedergegeben und in Material 8 ist das zur Kommunikation zugehörige Sequenzdiagramm auszugsweise dargestellt.

3.1 Analysieren Sie das RFC-Protokoll und beschreiben Sie das Sequenz-Diagramm und den daraus folgenden Ablauf in der Software.

(10 BE)

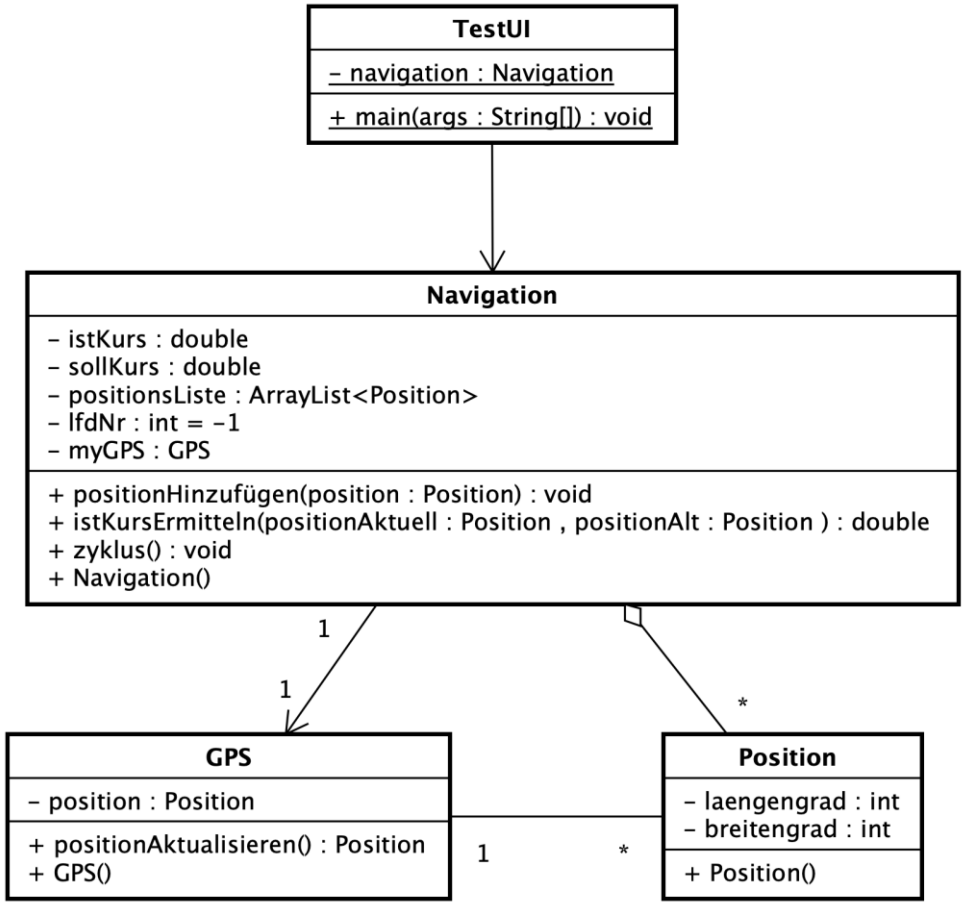
3.2 Entwickeln Sie das vollständige Struktogramm für die Operation `holeAnfrage()` in dem Objekt `wind`.

Hinweis: Es ist das das RFC-Protokoll zu berücksichtigen.

(8 BE)

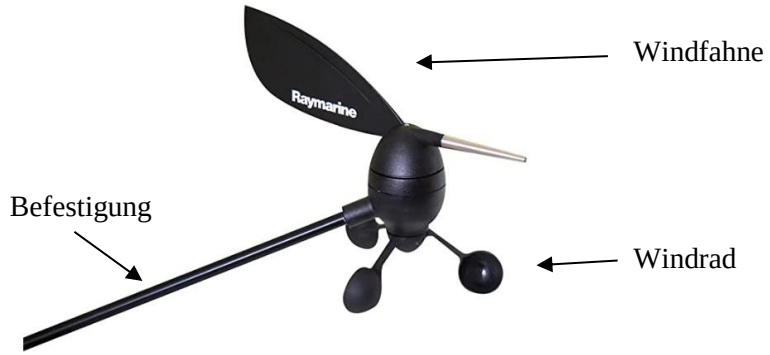
Material 1

UML-Klassendiagramm



Material 2

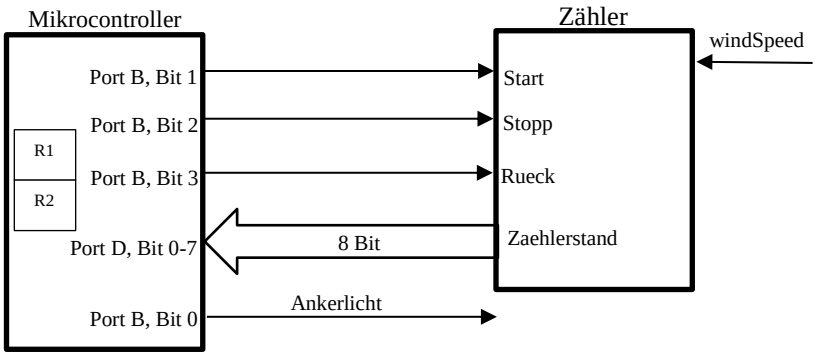
Windsensorsystem



https://www.amazon.de/RAYMARINE-ST60-LONG-MASTHEAD-TRANSDUCER/dp/B000WOJ7RQ/ref=sr_1_5?__mk_de_DE=ÅMÅŽÕÑ&dchild=1&keywords=Raymarine+ST60&qid=1629189865&sr=8-5 (abgerufen am 17.8.2021).

Material 3

Schaltungsaufbau für einen Mikrocontroller mit Zähler

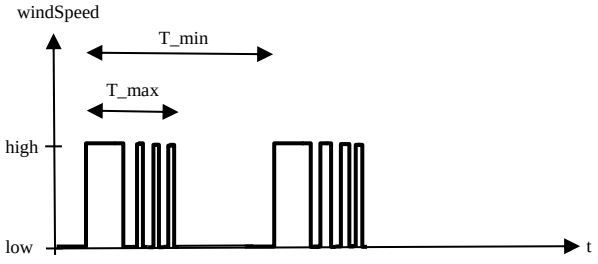


Material 4

Signal-Zeit-Diagramm des Ausgangssignals des Windrads

Material 5

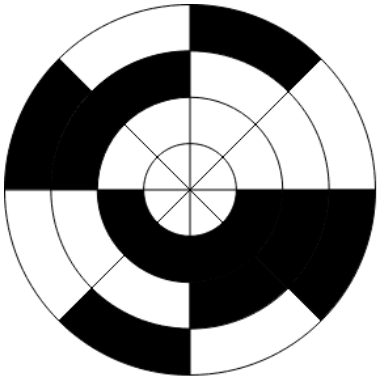
Signal-Zeit-Diagramm eines anderen möglichen Ausgangssignals des Windrads



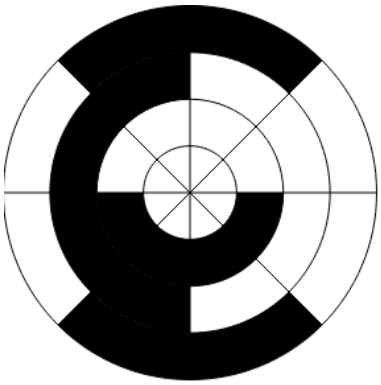
Material 6

Codescheiben im Binärcode und im Gray-Code

Codescheibe Binärcode



Codescheibe Gray-Code



https://slidetodoc.com/egr-2131-unit-8-encoders-code-converters-and/ (abgerufen am 04.10.2021).

Material 7

RFC-Protokoll

RFC – Ausschnitt für das Protokoll zwischen Server und Client		
DatenServer Server	**WindSensor Client**	**Bedeutung**
…		
sendet String: sendWindSpeed		Der Server fordert den Client auf, die Geschwindigkeit des Windrads in Umdrehungen pro Sekunde zu übermitteln
	sendet int: Umdrehungen	Der Client liefert die Umdrehungen pro Sekunde
sendet String: setAnkerLicht		Der Server übermittelt den Status des Ankerlichtes
sendet int: Status		(1 = an, 0 = aus)
	sendet String: ack	Der Client bestätigt den Erhalt
sendet String: getErrorCode		Server fragt Fehlercode ab
	sendet int: errorCode	Client sendet den ErrorCode 0 = „alles ok“, 1 = „Fehler“
…		

Material 8

Sequenzdiagramm